Curtin University — Discipline of Computing

Mobile Application Development (COMP2008)

Mock Examination

Attempt this mock exam paper in preparation for the final examination. Answer all questions by yourself.

Warning

The purpose of this mock exam is to give you some idea of what to expect, *not* to be a guide to the unit material. While the real exam is likely to follow roughly the same format, and cover the same broad topics, the questions will of course be different. To answer the real exam successfully, it is vital that you understand how to solve the problems posed in the practical worksheets.

The real exam will also be closed book – no books, notes, electronic devices, etc.

Will you upload the answers?

No. Sample answers to this mock exam *will not* be provided – no exceptions.

Why?

Sample answers discourage people from putting in real effort to learn the concepts and skills. They encourage rote (fake) learning, where you try to memorise an answer without understanding how to obtain it or even why it's correct.

Basically, if you're given the answers, it's too easy to convince yourself that you don't need to work them out.

Updated: 7th November, 2018

Question 1

(a) Explain the difference between the following pairs of XML layout attributes:

(i) `android:layout_constraintTop_toTopOf="parent"` and `android:layout_constraintTop_toBottomOf="@id/box"`.

(ii) `android:layout_height="0dp"` and `android:layout_height="1dp"`.

(b) Given a `<Button .../>` XML element, at what point does a corresponding `Button` object come into existence?

(c) For each of the following statements, state whether you agree or disagree, and explain why:

(i) *"Every view must have an ID, or else it cannot be represented as an object."*

(ii) *"A view's ID does not always need to be unique across the whole user interface."*

(d) Say you want your app to have different UI layouts for portrait and landscape orientations, as well as a third one where the screen is landscape and at least 500 dp units high. What needs to be done in terms of the layout XML files?

Question 2

(a) Consider the following method:

```
@Override
public void onActivityResult(int request, int result, Intent intent)
{
    ...
}
```

What is this particular `Intent` object used for, and where is it initialised?

(b) Say you want to add a fragment to an activity (in the activity's `onCreate()` method).

(i) Explain why the fragment may *already exist* (and hence why you should check for it first).

(ii) If the fragment does not already exist, what must you do?

Question 2 continues on the next page

(c) Suppose we want to display a list of strings using RecyclerView, with a button for each string that will convert it to uppercase. So far we have the following partial implementation:

```
public class MyFragment extends Fragment
{
    private String[] dataList = new String[100];
    private MyAdapter myAdapter;

    ... // Other fields, methods and classes as required.

    private class MyViewHolder extends RecyclerView.ViewHolder
    {
        private int index;
        private TextView text;
        private Button btn;

        public MyViewHolder(LayoutInflater li, ViewGroup p)
        {
            super(li.inflate(R.layout.my_list, p, false));
            text = itemView.findViewById(R.id.text);
            btn = itemView.findViewById(R.id.btn);
            // (i) What goes here?
        }

        public void bind(String data)
        {
            this.index = getAdapterPosition();
            // (ii) What goes here?
        }
    }
}
```

Provide the code required at (i) and (ii). Make reasonable assumptions, if necessary, about the spelling of relevant API classes and methods.

(Hint: the correct answer makes use of *all* the fields shown above.)

Question 3 appears on the next page

Question 3

(a) Consider the following:

```
public class ProjectCursor extends CursorWrapper
{
    public ProjectCursor(Cursor c)
    {
        super(c);
    }

    public Project getProject()
    {
        ...
    }
}
```

Explain what the `getProject()` method needs to do. Without worrying about the precise details of a `Project` object, be as specific as you can in your answer.

(b) To update a database table row using the `SQLiteDatabase.update()` method, what information do you need, and *in what form / object structure* do you need it?

(c) After invoking the contacts app to have the user pick a contact, your app *will not* immediately be told which contact has been selected. What information will it receive instead, and what must it do to actually access the contact's name and ID?

(d) Give one example of each of the following:

(i) A situation where your app must request permission to access information.

(ii) A situation where your app must grant permission to another app.

Question 4 appears on the next page

Question 4

(a) Consider the following model classes:

```
public class Volcano
{
    private String name;
    private int height;
    private int[] eruptionYears;
    ...
}
```

```
public class SurveyData
{
    private List<Volcano> volcanoes;
    private String lastUpdated;
    ...
}
```

Give an example to show how this data could be (sensibly) encoded in a JSON form.

(b) What role does `HttpURLConnection` play in downloading a web-based resource? Also explain *where* would you use it in a mobile app.

(c) Why is HTTPS especially important when connecting to web services?

(d) Briefly describe TWO (2) different problems that may arise when connecting to a web service (even assuming the client-side code is bug-free).

Question 5 appears on the next page

Question 5

(a) For each of the following statements, state whether you agree or disagree, and explain why:

   (i) *"The TypeScript compiler helps you avoid certain kinds of bugs."*

   (ii) *"Unlike mobile apps, the client-side part of a web app cannot directly contact just any web service."*

(b) Give CSS selectors for matching the following sets of HTML/DOM elements:

   (i) All `<input>` elements with a value of "`abracadabra`".

   (ii) All `<em>` elements with class "real" that occur inside `<div>` elements with class "fancy".

(c) Using jQuery, set up an event handler for a "Reset" button that, when pressed, sets the value of all text fields to `""`. Assume the reset button itself has an ID of "reset". Assume that by "text fields" we mean `<input>` elements of type `text`.

(d) How do promise objects simplify error handling in asynchronous JavaScript code?

End of Mock Test